Proof planning Non-standard Analysis

Proof-planning Non-standard Analysis

Ewen Maclean, Jacques Fleuriot and Alan Smaill
{E.Maclean,J.Fleuriot,A.Smaill}@ed.ac.uk
Division of Informatics
University of Edinburgh
80 South Bridge

November 28, 2001

1 Introduction

This paper presents work carried out in the $\lambda Clam$ proof-planner (Richardson *et al.* 00) on automating mathematical proofs using induction and non-standard analysis. The central idea is to show that the proofs we present are well-suited to proof-planning, due to their shared common structure. The theorems presented in this paper belong to standard analysis, and have been proved using induction and techniques from non-standard analysis.

We first give an overview of the proof-planning paradigm, giving a brief exposition of rippling as a heuristic for guiding rewriting. We then present the basic notions of non-standard analysis and explain our axiomatisation. We then go on to explain the theorems we intend to prove and sketch their proofs. Finally we show the parts of the proofs which have been planned automatically in $\lambda Clam$ and draw some conclusions from the work completed so far.

2 Proof-planning

Proof planning (Bundy 88) is a technique for devising an overall plan for a proof, which can then be used to guide the proof search itself. A proof-plan is made up of a series of applications of methods. The methods embody common patterns within proofs, such as the use of induction, and the associated tactics carry out these methods explicitly in the object level prover. For each conjecture a precise proof-plan is built, but a general form of a proof-plan can be developed for certain classes of conjecture. As described by (Bundy 88), a method in proof planning has a number of "slots" assigned to it which correspond to a name by which it is recognised, an input which is matched against the current goal, and a set of preconditions which determine whether the method is applicable to the current goal. The method also has a set of effects which define what happens to the goal after it has been applied, and a tactic which controls the object level prover, in which the plan can be executed. Finally, it has an output which is the result of applying the effects to the current goal.

2.1 Rippling

Rippling is a heuristic used in proof-planning for guiding the proof search. It was initially motivated by an observation on how terms introduced by induction are affected by rewriting. Bundy formalised this idea in (Bundy *et al.* 93), and a formal calculus has been developed from which one can prove termination (Basin & Walsh 96).

Rippling annotates the conclusion to indicate the difference between the hypotheses and conclusion. The idea is that the extra structure which exists in the conclusion can be annotated, and one can use this annotation to reason about how the proof is proceeding. For example, a conclusion $s(x+y) = y + s(x)$, could be annotated as

$$\boxed{s(\boxed{x+y})}^{\uparrow} = y + \boxed{s(\boxed{x})}^{\uparrow}$$

with respect to the hypothesis $x + y = y + x$. Here, the hypothesis is contained within the "holes" and the unannotated parts of the conjecture, and the arrow means that the proof is current trying to move the successor function outwards. The way in which the hypothesis occurs within the conclusion is called its *embedding*. We refer to these annotated rewrite rules as *wave rules*.

3 Non-standard analysis

Non-standard analysis is a tool which provides an intuitive yet rigorous alternative to Weierstraß's tricky ϵ-δ proofs. Abraham Robinson brought together a number of ideas from mathematical logic to come up with a theory of analysis in the hyperreal domain – a proper field extension of the reals (Robinson 66). By formally introducing an "infinitely close" relation, and numbers systems such as the hyperreals (${}^*\mathbb{R}$) and the the hypernaturals (${}^*\mathbb{N}$), definitions and proofs become simpler and more intuitive.

We illustrate our last remark by considering the standard and the non-standard formulations of a limit:

Definition 1 *The limit l, of a function f at a point a is defined as:*

$$\lim_{x\to a} f(x) = l \quad \equiv \quad \forall \epsilon \in \mathbb{R}.\ \epsilon > 0 \to \exists \delta \in \mathbb{R}.\ \delta > 0 \to \forall x \in \mathbb{R}.\quad 0 < |x - a| < \delta \to |f(x) - l| < \epsilon$$

Proofs which make use of this definition are usually difficult to automate, on account of the alternating quantifiers, which mean that instantiations for δ have to guessed early on in the proof. Theorem 1 gives us an alternative characterisation of a limit making use of the infinitely close relation $\approx$.

Theorem 1 *The limit l, of a function f at a point a can equivalently be defined in non-standard analysis by:*

$$\lim_{x\to a} f(x) = l \quad \iff \quad \forall x \in {}^*\mathbb{R}.\ x \approx \hat{a} \wedge x \neq \hat{a} \to {}^*f(x) \approx \hat{l}.$$

As this theorem expresses the notion of limit over the hyperreals, a few points are worth mentioning: the function ${}^*f$ is the original function f extended to accept hyperreal arguments whereas $\hat{a}$ corresponds to the embedding of real value a in the hyperreals. These notions are introduced for correctness and will not be examined further here. Only an intuitive understanding of their behaviour is needed for the rest of this paper although the interested reader may find out more by examining (Robinson 66).

This theorem gives us a simple non-standard characterisation for a limit which formalises the intuition that underlies the ϵ-δ formulation of a limit. For a proof of theorem 1 the reader is urged to consult (Fleuriot & Paulson 00), for example.

4 The Methodology

This paper presents theorems of standard analysis, formulated in a non-standard setting. By using the definitions of non-standard analysis, as presented in section 3, the proofs of these theorems become algebraic and hence easier to automate. Proof-planning provides a technique for capturing common patterns of reasoning. The proofs we present contain such reasoning patterns. We devise an approximating recursive function (the " auxiliary function") about which lemmas can be proved that then lead to an algebraic proof of the original goal (see section 5.2 for more details).

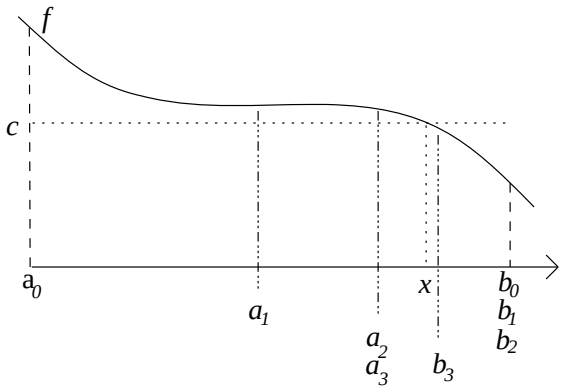

Figure 1: The sequence of partitions

As an example, let us consider the case of the intermediate value theorem for example. We have a function f which is continuous on a region $[a, b]$, where $f(a) > f(b)$, and we are required to show that for any point c, for which $f(a) \leq c \leq f(b)$, there is some point x such that $f(x) = c$. A way of proving that such a point exists, is to partition the original region, by continually halving it, and choosing the left interval if the value of the function at the current partition point is less than c, and otherwise choosing the right. In figure 4, the interval which corresponds to the result of the ith recursive call of the auxiliary function is labelled $[a_i, b_i]$. The base case interval, $[a_0, b_0]$ is the entire domain over which we reason about the function f, i.e. $[a, b]$. By choosing a criterion for whether to choose the left or right half of the current interval, the auxiliary function builds a sequence of intervals, at whose endpoints the function f converges to c.

We can now prove that the point c always exists within any interval $[f(a_i), f(b_i)]$, and we can show that the length of the interval $[a_i, b_i]$ is always $\frac{b-a}{2^i}$. The proofs of these intermediate lemmas are done using structural induction over the standard natural numbers. Since we have formulated the theorem in a non-standard setting,

we can evaluate the lemmas at an infinite hypernatural, and show that in such a case only one real number can exist which still lies within the now infinitesimal interval, namely x.

In standard analysis, an ϵ-δ formulation is used to find the limit of the sequence a_i and b_i as i tends to infinity. In non-standard analysis we can avoid the complicated notions of limits and instead evaluate the sequence at a infinite hypernatural, and use the simplified definitions to get an algebraic proof.

4.1 Transfer

The transfer principle for real analysis consists of a definition of the *-transform, a transformation which converts sentences over the real numbers to sentences over the hyperreal numbers, and a theorem demonstrating the equivalence between the two representations:

Definition 2 *If ϕ is a first order statement expressed over the reals, then the *-transform of ϕ denoted ${}^*\phi$ is defined inductively by the following rules:*

replace functions and predicates in ϕ by their non-standard extensions;

replace unquantified real numbers in ϕ by their embeddings in the hyperreals;

variables quantified over the reals in ϕ become quantified over the hyperreals in ${}^*\phi$.

Theorem 2 *Any true first order statement ϕ expressed over the reals, $\mathbb{R}$, is true if and only if its transform ${}^*\phi$ is true over the hyperreals, ${}^*\mathbb{R}$.*

Now Consider the statement of the Intermediate Value Theorem:

Theorem 3 *Let f be a function which is continuous on the closed interval [a, b]. Suppose that c is a real number between f(a) and f(b); then there exists x in [a, b] such that f(x) = c.*

We must consider how to state this theorem in non-standard analysis. In some text books, the c here is presented as a universally quantified variable in the conclusion, and in some it is a parameter. In the standard case these two steps are equivalent, but the transfer principle means the non-standard versions of these two statements are different.

In the first approach, where the standard conclusion is:

$$\forall c \in \mathbb{R}\ \exists x \in \mathbb{R}\ f(x) = c$$

the *-transformed conclusion is:

$$\forall c \in {}^*\mathbb{R}\ \exists x \in {}^*\mathbb{R}\ {}^*f(x) = c$$

and in the second approach where the standard conclusion is:

$$\exists x \in \mathbb{R}\ f(x) = c$$

the non-standard conclusion is:

$$\exists x \in {}^*\mathbb{R}\ {}^*f(x) = \hat{c}$$

where c is now of type `real`.

The transfer principle tells us that if any one of the transferred versions is true, then the standard version must also be true. The problem is that the proof of one may rely on the proof of the standard version and the transfer principle, while the other may be provable, reasoning purely in the non-standard domain.

The above observation is important as it provides a justification for non-standard characterisations. In the current work, as can be seen in section 5.2 we opt for the second approach when specifying non-standard versions of familiar theorems.

5 Case Study

We present two theorems of standard analysis, and show how they are stated using simplified definitions from non-standard analysis. We give a detailed description of the first proof, showing specific extracts from the proof-planner $\lambda Clam$ where necessary.

5.1 The system

The real numbers are dealt with in $\lambda Clam$ by providing the planner with the field axioms. These axioms are also true for the hyperreals, and there are certain other axioms which connect the two number systems, the most important to this work being:

$$\forall x, y \in \mathbb{R}.\ \ \hat{x} \approx \hat{y} \rightarrow x = y \tag{1}$$

$$^*f(\hat{x}) = \widehat{f(x)} \tag{2}$$

These state that only one real number can exist within an infinitesimal region, and that functions over the reals are well-defined in the hyperreals.

In the excerpts from the planner given in section 5.2, the field axioms for the reals and hyperreals are given as axioms to the system. The rules which have been proved separately by the system, which we use for the proofs presented here, are:

$$\frac{X}{Y \times Z} \quad \Rightarrow \quad \frac{1}{Y} \times \frac{X}{Z} \tag{3}$$

$$X - \frac{X+Y}{2} \quad \Rightarrow \quad \frac{X-Y}{2} \tag{4}$$

$$\frac{X+Y}{2} - Y \quad \Rightarrow \quad \frac{X-Y}{2} \tag{5}$$

and we add the usual recursive definition for exponentiation:

$$X^{s(Y)} \Rightarrow X \times X^Y \tag{6}$$

$$\tag{7}$$

As each of the proofs involves an operation which continually halves the partition, these rules are used often. This set of rules can be interpreted as encapsulating a common pattern of reasoning in these proofs, and as such constitute part of the Proof-planning machinery.

5.2 The intermediate value theorem

The intermediate value theorem can be stated in non-standard analysis as:

$$f : \mathbb{R} \rightarrow \mathbb{R} \tag{8}$$

$$a, b, c : \mathbb{R} \tag{9}$$

$$\forall x, y \in {}^*\mathbb{R}.\ \ \hat{a} \leq x \leq \hat{b}\ \wedge\ \hat{a} \leq y \leq \hat{b}\ \wedge\ x \approx y\ \rightarrow\ {}^*f(x) \approx {}^*f(y) \qquad \texttt{(continuity)} \tag{10}$$

$$a < b \tag{11}$$

$$f(a) \leq c \leq f(b) \tag{12}$$

$$\vdash \exists x \in \mathbb{R}.\ a \leq x \leq b \rightarrow f(x) = c \tag{13}$$

We present a proof for this conjecture using the methodology outlined in section 4.

5.2.1 Definition of recursive function

We define the auxiliary recursive function, `ivtrec` : $(\mathbb{R} \rightarrow \mathbb{R}) \times \mathbb{R} \times \mathbb{R} \times \mathbb{R} \times \mathbb{N} \rightarrow (\mathbb{R} \times \mathbb{R})$, which returns a pair representing the interval $[A, B]$ as follows:

```
ivtrec F A B C 0 = [A,B]

ivtrec F A B C s(N) =  (let [X,Y]=ivtrec F A B C N
                        in if F((X+Y)/2)<=C then [(X+Y)/2,Y]
                                            else [X,(X+Y)/2])
```

which can be expressed as rewrite rules by splitting the result up into the left and right endpoints of the interval. By annotating the rules we arrive at the following wave rules:

$$\texttt{ivtrec}_\texttt{l}\ F\ A\ B\ C\ 0 \qquad \Rightarrow A$$

$$\texttt{ivtrec}_\texttt{r}\ F\ A\ B\ C\ 0 \qquad \Rightarrow B$$

$F((\texttt{ivtrec}_\texttt{l}\ F\ A\ B\ C\ N + \texttt{ivtrec}_\texttt{r}\ F\ A\ B\ C\ N)/2) > C \rightarrow \quad \texttt{ivtrec}_\texttt{l} F\ A\ B\ C\ \boxed{s(N)}^{\uparrow} \quad \Rightarrow \texttt{ivtrec}_\texttt{l}\ F\ A\ B\ C\ N$

$F((\texttt{ivtrec}_\texttt{l}\ F\ A\ B\ C\ N + \texttt{ivtrec}_\texttt{r}; F\ A\ B\ C\ N)/2) > C \rightarrow \quad \texttt{ivtrec}_\texttt{r} F\ A\ B\ C\ \boxed{s(N)}^{\uparrow} \quad \Rightarrow \boxed{(\texttt{ivtrec}_\texttt{r}\ F\ A\ B\ C\ N} + \texttt{ivtrec}_\texttt{l}\ F\ A\ B\ C\ N)/2^{\uparrow}$

$F((\texttt{ivtrec}_\texttt{l}\ F\ A\ B\ C\ N + \texttt{ivtrec}_\texttt{r}\ F\ A\ B\ C\ N)/2) \leq C \rightarrow \quad \texttt{ivtrec}_\texttt{l} F\ A\ B\ C\ \boxed{s(N)}^{\uparrow} \quad \Rightarrow \boxed{(\texttt{ivtrec}_\texttt{l}\ F\ A\ B\ C\ N} + \texttt{ivtrec}_\texttt{r}\ F\ A\ B\ C\ N)/2^{\uparrow}$

$F((\texttt{ivtrec}_\texttt{l}\ F\ A\ B\ C\ N + \texttt{ivtrec}_\texttt{r}\ F\ A\ B\ C\ N)/2) \leq C \rightarrow \quad \texttt{ivtrec}_\texttt{r} F\ A\ B\ C\ \boxed{s(N)}^{\uparrow} \quad \Rightarrow \texttt{ivtrec}_\texttt{r}\ F\ A\ B\ C\ N$

In the output of $\lambda Clam$ the name of the set of wave rules which corresponds to rewriting $\texttt{ivtrec}_\texttt{l}$ is `ivt_l`, and the name of the set of wave rules which corresponds to rewriting `ivtrec_r` is $\texttt{ivt}_\texttt{r}$.

5.2.2 Lemma 1

The first lemma we tackle is:

$$\forall n \in \mathbb{N}.\ \texttt{ivtrec}_\texttt{r}\ f\ a\ b\ c\ n\ - \texttt{ivtrec}_\texttt{l}\ f\ a\ b\ c\ n = \frac{b-a}{2^n}$$

We use a standard structural induction scheme for the natural numbers, to set up an approximation result which is valid also in the non-standard domain because of the transfer principle. The resulting non-standard formulation is:

$$\forall n \in {}^*\mathbb{N}.\ {}^*\texttt{ivtrec}_\texttt{r}\ {}^*f\ \hat{a}\ \hat{b}\ \hat{c}\ n\ - {}^*\texttt{ivtrec}_\texttt{l}\ {}^*f\ \hat{a}\ \hat{b}\ \hat{c}\ n = \frac{\hat{b}-\hat{a}}{2^n}$$

We use this to show that for an infinite n, the interval defined by `ivtrec` is infinitely small. The base case to the proof of the lemma is:

$$\texttt{ivtrec}_\texttt{r}\ f\ a\ b\ c\ 0 - \texttt{ivtrec}_\texttt{l}\ f\ a\ b\ c\ 0 = b - a$$

which is trivially proved by rewriting and identity, as can be seen in the following output from $\lambda Clam$:

```
a:ℝ,f:ℝ → ℝ,b:ℝ,c:ℝ
⊢ ivtrecr (f, a, b, c, 0) - ivtrecl (f, a, b, c, 0) =  (b - a)/2^0

Attempting...
Method application:  rewrite_with ivt_r
succeeded
a:ℝ,f:ℝ → ℝ,b:ℝ,c:ℝ
⊢ b - ivtrecl (f, a, b, c, 0)) = (b - a)/2^0

Attempting...
Method application:  rewrite_with ivt_l
succeeded
a:ℝ,f:ℝ → ℝ,b:ℝ,c:ℝ
⊢ b - a = (b - a)/2^0

Attempting...
Method application:  rewrite_with exp1
succeeded
a:ℝ,f:ℝ → ℝ,b:ℝ,c:ℝ
⊢ b - a = (b - a)/1
Attempting...

Method application:  rewrite_with invtimes_ident
succeeded
a:ℝ, f:ℝ → ℝ, b:ℝ, c:ℝ
⊢ b - a = (b - a) × 1

Attempting...
Method application:  rewrite_with times_ident
succeeded
a:ℝ,f:ℝ → ℝ,b:ℝ
c:ℝ
```

```
⊢ b - a = b - a

Attempting...
Method application:  rewrite_with refleq
succeeded
a:ℝ,f:ℝ → ℝ,b:ℝ,c:ℝ
⊢ trueP

trueGoal!
branch closed!
```

The step case becomes:

$$\texttt{ivtrec}_\texttt{r}\ f\ a\ b\ c\ \boxed{s(n)}^{\uparrow} - \texttt{ivtrec}_\texttt{l}\ f\ a\ b\ c\ \boxed{s(n)}^{\uparrow} = \frac{b-a}{2^{\boxed{s(n)}^{\uparrow}}}$$

In $\lambda Clam$ the induction method first finds an embedding, so rippling can take place:

```
Method application:  induction_meth nat_struct
succeeded

Method application:  step_case
succeeded

Method application:  set_up_ripple
succeeded
a:ℝ,f:ℝ → ℝ,b:ℝ,c:ℝ,N:ℕ
ivtrecr (f, a, b, c, N) - ivtrecl (f, a, b, c, N) = (b - a)/2^N
⊢ ivtrecr (f, a, b, c, s N) - ivtrecl (f, a, b, c, s N)) = (b - a)/2^s(N)
```

After the application of arithmetical rules 3 and 6, and the wave rules for $\texttt{ivtrec}_\texttt{l}$ and $\texttt{ivtrec}_\texttt{r}$, the conjecture splits into four cases. The conditions to the first case are:

$$f((\texttt{ivtrec}_\texttt{l}\ f\ a\ b\ c\ n + \texttt{ivtrec}_\texttt{r}\ f\ a\ b\ c\ n)/2) \leq c\ \wedge\ f((\texttt{ivtrec}_\texttt{l}\ f\ a\ b\ c\ n + \texttt{ivtrec}_\texttt{r}\ f\ a\ b\ c\ n)/2) \leq c$$

and the conclusion becomes:

$$\boxed{\frac{\texttt{ivtrec}_\texttt{l}\ f\ a\ b\ c\ n + \boxed{\texttt{ivtrec}_\texttt{r}\ f\ a\ b\ c\ n}}{2}}^{\uparrow} - \texttt{ivtrec}_\texttt{l}\ f\ a\ b\ c\ n = \boxed{\frac{1}{2} \times \boxed{\frac{b-a}{2^n}}}^{\uparrow}$$

which can be rewritten using arithmetic rewrite rules 4 and 5 to:

$$\boxed{\frac{1}{2} \times \boxed{\texttt{ivtrec}_\texttt{r}\ f\ a\ b\ c\ n - \texttt{ivtrec}_\texttt{l}\ f\ a\ b\ c\ n}}^{\uparrow} = \boxed{\frac{1}{2} \times \boxed{\frac{b-a}{2^n}}}^{\uparrow}$$

Now the proof is completed by rewriting with the induction hypothesis, which is called "weak fertilisation" in Proof-planning terminology. The resulting goal is an identity:

```
Attempting...
Method application:  fertilise
succeeded
N:ℕ,a:ℝ, f:ℝ → ℝ, b:ℝ, c:ℝ
f((ivtrecl (f, a, b, c, N) + ivtrecr (f, a, b, c, N))/2) > c
ivtrecr (f, a, b, c, N) - ivtrecl (f, a, b, c, N)) = (b - a)/2^N
⊢ 1/2 × (ivtrecr (f, a, b, c, N) - ivtrecl (f, a, b, c, N)) =
  1/2 × (b - a)/2^N

Attempting...
```

```
fertilisation_strong
failed

Attempting...
Method application:  fertilisation_weak
succeeded
N:ℕ,a:ℝ, f:ℝ → ℝ, b:ℝ, c:ℝ
f((ivtrecl (f, a, b, c, N) + ivtrecr (f, a, b, c, N))/2) > c
⊢ 1/2 × (b - a)/2^N = 1/2 × (b - a)/2^N
```

The other cases of the proof follow similarly.

5.2.3 Lemma 2

The second lemma to tackle is:

$$\forall n \in \mathbb{N}.\ \texttt{ivtrec}_\texttt{r}\ f\ a\ b\ c\ s(n)\ \leq\ \texttt{ivtrec}_\texttt{r}\ f\ a\ b\ c\ n \wedge \mathit{ivtrec}_l\ f\ a\ b\ c\ s(n)\ \geq\ \texttt{ivtrec}_\texttt{l}\ f\ a\ b\ c\ n$$

This lemma is solved in a similar way to lemma 1, so we omit the proof here.

5.2.4 Lemma 3

The third Lemma we must solve is:

$$\forall n \in \mathbb{N}.\ f(\texttt{ivtrec}_\texttt{r}\ f\ a\ b\ c\ n)\ \geq\ c \wedge c\ \geq\ f(\texttt{ivtrec}_\texttt{l}\ f\ a\ b\ c\ n)$$

which is a more complicated lemma to solve using induction. The base case becomes:

$$\texttt{ivtrec}_\texttt{r}\ f\ a\ b\ c\ 0\ \geq\ c \wedge c\ \geq\ \texttt{ivtrec}_\texttt{l}\ f\ a\ b\ c\ 0$$

which reduces to one of the hypotheses after rewriting. As the goal is a conjunction we show just the conjunct involving $\texttt{ivtrec}_\texttt{r}$:

```
Method application:  sym_eval
succeeded
a:ℝ,f:ℝ → ℝ,b:ℝ,c:ℝ
f(b) = c ∨ f(b)> c
⊢ f(ivtrecr (f, a, b, c, 0)) = c ∨ f(ivtrecr (f, a, b, c, 0)) > c

Attempting...
Method application:  rewrite_with ivt_r
succeeded
a:ℝ,f:ℝ → ℝ,b:ℝ,c:ℝ
f(b) = c ∨ f(b)> c
⊢ f(b) = c ∨ f(ivtrecr (f, a, b, c, 0)) > c

Attempting...
Method application:  rewrite_with ivt_r
succeeded
a:ℝ,f:ℝ → ℝ,b:ℝ,c:ℝ
f(b) = c ∨ f(b)> c
⊢ f(b) = c ∨ f(b) > c
```

The step case conclusion becomes:

$$\forall n \in \mathbb{N}.\ \texttt{ivtrec}_\texttt{r}\ f\ a\ b\ c\ \boxed{s(n)}^{\uparrow}\ \geq\ c \wedge c\ \geq\ \texttt{ivtrec}_\texttt{l}\ f\ a\ b\ c\ \boxed{s(n)}^{\uparrow}$$

which again produces two separate proofs each with two cases. We study just the case where we rewrite $\texttt{ivtrec}_\texttt{l}$ with the condition

$$f((\texttt{ivtrec}_\texttt{l}\ f\ a\ b\ c\ n + \texttt{ivtrec}_\texttt{r}\ f\ a\ b\ c\ n)/2) > c$$

the conclusion becomes

$$c \geq \uparrow(\texttt{ivtrec}_\texttt{l}\ f\ a\ b\ c\ n + \texttt{ivtrec}_\texttt{r}\ f\ a\ b\ c\ n)/2$$

which is represented by the following application of wave rules in $\lambda Clam$:

```
Method application:   wave_case_split ivt_r
succeeded
f(ivtrecr (f, a, b, c, N)) =  c ∨ f(ivtrecr (f, a, b, c, N)) > c
f(b) = c ∨ f(b) > c
f((ivtrecl (f, a, b, c, N) + ivtrecr (f, a, b, c, N))/2) > c
⊢ f(ivtrecr (f, a, b, c, s N)) = c ∨ f(ivtrecr(f, a, b, c, s N)) > c

Method application:  wave_method inout ivt_r
succeeded
f(ivtrecr (f, a, b, c, N)) =  c ∨ f(ivtrecr (f, a, b, c, N)) > c
f(b) = c ∨ f(b) > c
f((ivtrecl (f, a, b, c, N) + ivtrecr (f, a, b, c, N))/2) > c
⊢ f((ivtrecl (f, a, b, c, N) + ivtrecr (f, a, b, c, N))/2) = c ∨
  f(ivtrecr(f, a, b, c, s N)) > c

Method application:  wave_method inout ivt_r
succeeded
f(ivtrecr (f, a, b, c, N)) =  c ∨ f(ivtrecr (f, a, b, c, N)) > c
f(b) = c ∨ f(b) > c
f((ivtrecl (f, a, b, c, N) + ivtrecr (f, a, b, c, N))/2) > c
⊢ f((ivtrecl (f, a, b, c, N) + ivtrecr (f, a, b, c, N))/2) = c ∨
  f((ivtrecl (f, a, b, c, N) + ivtrecr (f, a, b, c, N))/2) > c
```

which is now subsumed by the condition, and hence the lemma is proved.

5.2.5 Finishing the proof

The results we have managed to prove so far can be transferred to the non-standard domain, and can then be expressed as follows:

$$\forall n \in {}^*\mathbb{N}.\ {}^*\texttt{ivtrec}_\texttt{r}\ {}^*f\ \hat{a}\ \hat{b}\ \hat{c}\ n\ -\ {}^*\texttt{ivtrec}_\texttt{l}\ {}^*f\ \hat{a}\ \hat{b}\ \hat{c}\ n = \tfrac{\hat{b}-\hat{a}}{2^n} \tag{14}$$

$$\forall n \in {}^*\mathbb{N}.\ {}^*\texttt{ivtrec}_\texttt{r}\ {}^*f\ \hat{a}\ \hat{b}\ \hat{c}\ s(n) \leq {}^*\texttt{ivtrec}_\texttt{r}\ {}^*f\ \hat{a}\ \hat{b}\ \hat{c}\ n \wedge {}^*\texttt{ivtrec}_\texttt{l}\ {}^*f\ \hat{a}\ \hat{b}\ \hat{c}\ s(n) \geq {}^*\texttt{ivtrec}_\texttt{l}\ {}^*f\ \hat{a}\ \hat{b}\ \hat{c}\ n \tag{15}$$

$$\forall n \in {}^*\mathbb{N}.\ {}^*f({}^*\texttt{ivtrec}_\texttt{r}\ {}^*f\ \hat{a}\ \hat{b}\ \hat{c}\ n) \geq \hat{c} \wedge \hat{c} \geq {}^*f({}^*\texttt{ivtrec}_\texttt{l}\ {}^*f\ \hat{a}\ \hat{b}\ \hat{c}\ n) \tag{16}$$

Now that we have these lemmas, we can show using (15) that at infinite n, $\texttt{ivtrec}_\texttt{l}\ {}^*f\ \hat{a}\ \hat{b}\ \hat{c}\ n \approx \texttt{ivtrec}_\texttt{r}\ {}^*f\ \hat{a}\ \hat{b}\ \hat{c}\ n$. We can use continuity (10) because its conditions are satisfied by (15) and (16), so at infinite n:

$${}^*f({}^*\texttt{ivtrec}_\texttt{l}\ {}^*f\ \hat{a}\ \hat{b}\ \hat{c}\ n) \approx \hat{c} \approx {}^*f({}^*\texttt{ivtrec}_\texttt{r}\ {}^*f\ \hat{a}\ \hat{b}\ \hat{c}\ n)$$

Now we also know from continuity that:

$$\forall y \in {}^*\mathbb{R}.\ \forall x \in \mathbb{R}.\ \ y \approx \hat{x} \rightarrow {}^*f(y) \approx {}^*f(\hat{x})$$

using these two formulas we have:

$${}^*f({}^*\texttt{ivtrec}_\texttt{l}\ {}^*f\ \hat{a}\ \hat{b}\ \hat{c}\ n) \approx \hat{c}$$
$${}^*f({}^*\texttt{ivtrec}_\texttt{l}\ {}^*f\ \hat{a}\ \hat{b}\ \hat{c}\ n) \approx {}^*f(\hat{x})$$

hence at a unique $x \in \mathbb{R}$ infinitely close to $\texttt{ivtrec}_\texttt{l}\ {}^*f\ \hat{a}\ \hat{b}\ \hat{c}\ n$, it must be the case that ${}^*f(\hat{x}) \approx \hat{c}$, and hence from (1) and (2) it must be the case that $f(x) = c$.

5.2.6 Notes on the proof

All of the main inductive lemmas mentioned in this proof were planned automatically in $\lambda Clam$. The planner was given lemmas (3),(4) and (5), which were all planned automatically. The final stage of the proof can be stated in the planner, but has not yet been planned completely.

5.3 Rolle's Theorem

We present an outline of a similar proof of Rolle's theorem. Rolle's theorem can be stated as:

$$
\begin{array}{rl}
f', f : \mathbb{R} \to \mathbb{R} & \\
a, b : \mathbb{R} & \\
\forall x, y \in {}^*\mathbb{R}.\ \hat{a} \le x \le \hat{b} \ \wedge\ \hat{a} \le y \le \hat{b} \ \wedge\ x \approx y \ \to\ {}^*f(x) \approx {}^*f(y) & (\texttt{continuity}) \\
\forall x \in \mathbb{R}.\ \forall h \in {}^*\mathbb{R}.\ a < x < b \ \wedge\ h \approx \hat{0} \ \wedge\ h \neq \hat{0} \ \to \frac{{}^*f(\hat{x}+h) - {}^*f(x)}{h} \approx {}^*f'(\hat{x}) & (\texttt{differentiability}) \\
a < b & \\
\vdash\ \exists c \in \mathbb{R}.\ a \le c \le b \ \to\ f'(c) = 0 &
\end{array}
$$

In order to solve this problem in a similar way to the Intermediate Value Theorem, it is necessary to introduce a case split, and to show that in each case a c can be found which satisfies the existentially quantified goal.

We perform a case split on the condition that either the function is constant in the region $[a, b]$, in which case the derivative is trivially zero everywhere on the region, or there is a maximum at some point c, or there is a minimum at some point c. If we can show that at these points the derivative must be zero, then that is enough to prove the existence of such a point in the original goal. In order to justify this case split, we use the fact that since f is continuous on a compact set, it must attain its maximum and minimum somewhere in that compact set. We focus here on the maximum case.

We define a recursive function to partition the region in the same way as in section 5.2. We can then prove the following theorems about the function:

$$\forall n \in \mathbb{N}.\ \texttt{rolrec}_\texttt{r}\ f\ f'\ a\ b\ n\ -\ \texttt{rolrec}_\texttt{l}\ f\ f'\ a\ b\ n = \tfrac{b-a}{2^n} \tag{17}$$

$$\forall n \in \mathbb{N}.\ \texttt{rolrec}_\texttt{r}\ f\ f'\ a\ b\ s(n)\ \le\ \texttt{rolrec}_\texttt{r}\ f\ f'\ a\ b\ n \wedge \texttt{rolrec}_\texttt{l}\ f\ f'\ a\ b\ s(n)\ \ge\ \texttt{rolrec}_\texttt{l}\ f\ f'\ a\ b\ n \tag{18}$$

$$\forall n \in \mathbb{N}.\ \texttt{rolrec}_\texttt{l}\ f\ f'\ a\ b\ n\ \le\ c\ \wedge\ \texttt{rolrec}_\texttt{l}\ f\ f'\ a\ b\ n \ge c \tag{19}$$

$$\forall n \in \mathbb{N}.\ f(\texttt{rolrec}_\texttt{l}\ f\ f'\ a\ b\ n) \le f(c)\ \wedge\ f(\texttt{rolrec}_\texttt{r}\ f\ f'\ a\ b\ n) \le f(c) \tag{20}$$

With these theorems proved the final result can be proved. As we transfer these results to the non-standard domain, and evaluate them at infinite hypernatural n, we get the following facts:

$$
\begin{array}{c}
{}^*\texttt{rolrec}_\texttt{l}\ {}^*f\ {}^*f'\ \hat{a}\ \hat{b}\ n \approx \hat{c} \approx {}^*\texttt{rolrec}_\texttt{r}\ {}^*f\ {}^*f'\ \hat{a}\ \hat{b}\ n \\
{}^*f({}^*\texttt{rolrec}_\texttt{l}\ {}^*f\ {}^*f'\ \hat{a}\ \hat{b}\ n) \approx {}^*f(\hat{c}) \approx {}^*f({}^*\texttt{rolrec}_\texttt{r}\ {}^*f\ {}^*f'\ \hat{a}\ \hat{b}\ n) \\
{}^*f({}^*\texttt{rolrec}_\texttt{l}\ {}^*f\ {}^*f'\ \hat{a}\ \hat{b}\ n) \le {}^*f(\hat{c}) \\
{}^*f({}^*\texttt{rolrec}_\texttt{r}\ {}^*f\ {}^*f'\ \hat{a}\ \hat{b}\ n) \le {}^*f(\hat{c})
\end{array}
$$

Now from the hypotheses we have:

$$\forall x \in \mathbb{R}.\ \forall h \in {}^*\mathbb{R}.\ a < x < b \ \wedge\ h \approx \hat{0} \ \wedge\ h \neq \hat{0} \ \to \frac{{}^*f(\hat{x}+h) - {}^*f(\hat{x})}{h} \approx {}^*f'(\hat{x})$$

we know that as ${}^*\texttt{rolrec}_\texttt{l}\ {}^*f\ {}^*f'\ \hat{a}\ \hat{b}\ n \approx {}^*f(\hat{c})$ we can write ${}^*\texttt{rolrec}_\texttt{l}\ {}^*f\ {}^*f'\ \hat{a}\ \hat{b}\ n$ as $x - h$ and we can also write ${}^*\texttt{rolrec}_\texttt{r}\ {}^*f\ {}^*f'\ \hat{a}\ \hat{b}\ n$ as $x + h$. Then we can deduce that $\hat{0} \le {}^*f'(\hat{c}) \le \hat{0}$, and since this must equal a real number, we can deduce that $f'(c) = 0$.

6 Related Work

The earliest work on automating proof in both standard analysis (Bledsoe *et al.* 72) and non-standard analysis (Bledsoe & Ballantyne 77) used a resolution based theorem prover. This work proved automatically many limit theorems in non-standard analysis. Though the proofs were automatic they were very difficult to read on account of the complicated resolution steps. Their prover did not attempt the theorems dealt with by the current work.

Since then many analysis proofs have been automated in the ΩMEGA proof-planner (Benzmüller *et al.* 97). The work is documented in (Melis *et al.* 00) and (Melis & Siekmann 99) for example. The focus here is on using knowledge-based proof-planning to prove limit theorems in standard analysis. The system introduces meta-variables during the proof when removing quantifiers, and instantiates them at the end of the proof using a constraint solver. Bledsoe's limit heuristic (Bledsoe *et al.* 72) is also incorporated.

The Mathpert system is a mathematical assistant which is also capable of automatically proving properties about functions in standard analysis such as continuity (Beeson 98). Many proofs involve side conditions on introduced variables which can be very complicated in the context of limits. Using non-standard analysis (Beeson 95) allows these conditions to be greatly simplified.

The ACL2 theorem prover did not include a theory for reasoning about conjectures over the reals, but Gamboa (Gamboa 99) used Nelson's Internal Set Theory (Nelson 77) to extend the prover to automatically prove theorems about real valued functions. This work presents a proof of the Intermediate Value Theorem, which uses similar reasoning steps to those presented here. The difference is the use of proof-planning and rippling to help complete the proof. It is unclear how much automation is needed to complete this proof in ACL2.

The most significant work that has been done in this area has been done in the interactive theorem prover Isabelle/HOL (Paulson 94), formalising the construction of the hyperreals, and proving a substantial amount of real analysis in the hyperreals using this construction (Fleuriot & Paulson 00). This work also mechanises a proof of Rolle's Theorem and of the Intermediate Value Theorem using non-standard definitions of continuity.

7 Conclusion

This work uses Proof-planning to capture common patterns of reasoning in the types of proof that we have presented. We build on the well developed research on Proof-planning for inductive theorems by extending the ideas to the challenging domain of real analysis. By avoiding the complicated definitions of limit of standard analysis, and replacing them with their algebraic counterparts from non-standard analysis, we arrive at more intuitive proofs, whose structure can be encapsulated by the techniques of Proof-planning.

References

David Basin and Toby Walsh. A calculus for and termination of rippling. *Journal of Automated Reasoning*, 16(1–2):147–180, 1996.

M. Beeson. Using nonstandard analysis to verify the correctness of computations. *International Journal of Foundations of Computer Science*, 6(3):299–338, 1995.

M Beeson. Automatic generation of epsilon-delta proofs of continuity. *Artificial Intelligence and Symbolic Computation*, pages 67–83, 1998. Lecture Notes in Artificial Intelligence No. 1476.

C. Benzmüller, L. Cheikhrouhou, D Fehrer, A. Fiedler, X. Huang, M. Kerber, K. Kohlhase, A Meier, E. Melis, W. Schaarschmidt, J. Siekmann, and V. Sorge. Ωmega: Towards a mathematical assistant. In W. McCune, editor, *14th International Conference on Automated Deduction*, pages 252–255. Springer-Verlag, 1997.

W. W. Bledsoe and A. M. Ballantyne. Automatic proofs of theorems in analysis using nonstandard techniques. *Association for Computing Machinery*, 24(3):353–374, 1977.

W. W. Bledsoe, R. S. Boyer, and W. H. Henneman. Computer proofs of limit theorems. *Artificial Intelligence*, 3:27–60, 1972.

A. Bundy. The use of explicit plans to guide inductive proofs. In R. Lusk and R. Overbeek, editors, *9th International Conference on Automated Deduction*, pages 111–120. Springer-Verlag, 1988. Longer version available from Edinburgh as DAI Research Paper No. 349.

A. Bundy, A. Stevens, F. van Harmelen, A. Ireland, and A. Smaill. Rippling: A heuristic for guiding inductive proofs. *Artificial Intelligence*, 62:185–253, 1993. Also available from Edinburgh as DAI Research Paper No. 567.

J. D. Fleuriot and L. C. Paulson. Mechanizing nonstandard real analysis. *LMS Journal of Computation and Mathematics*, 3:140–190, 2000.

R. Gamboa. *Mechanically Verifying Real-Valued Algorithms in ACL2.* Unpublished PhD thesis, The University of Texas at Austin, 1999.

E. Melis and J. Siekmann. Knowledge-based proof planning. *Artificial Intelligence*, 115(1):65–105, 1999.

E. Melis, J. Zimmer, and T. Müller. Extensions of constraint solving for proof planning. *European Conference on Artificial Intelligence*, 2000.

E. Nelson. Internal set theory: A new approach to nonstandard analysis. *Bulletin American Mathematical Society*, 83, 1977. Available from `http://www.math.princeton.edu/~nelson/books.html`.

L.C. Paulson. *Isabelle: A generic theorem prover*. Springer-Verlag, 1994.

J. Richardson, L. Dennis, J. Gow, and M. Jackson. User/programmer manual for the $\lambda Clam$ proof planner. 2000.

A. Robinson. *Non-standard Analysis*. North-Holland Publishing Company, Amsterdam, 1966. Studies in Logic and the Foundations of Mathematics.